

Docket No. AUS920010055US1

CLAIMS:

What is claimed is:

1. A method for preventing premature shutdown of a virtual machine, comprising the steps of:
- starting a waiter thread in the virtual machine;
- monitoring daemon threads running in the virtual machine by the waiter thread; and
- preventing shutdown of the virtual machine so long as any of the monitored daemon threads are running in the virtual machine.
2. The method as recited in claim 1, further comprising the steps of:
- registering daemon threads in a queue managed by the waiter thread.
3. The method as recited in claim 2, further comprising the step of:
- ~~responsive to a first daemon thread becoming~~
- inactive, searching for other inactive daemon threads registered in the queue.
4. The method as recited in claim 2, further comprising the step of:
- responsive to a new daemon thread being created, appending the new daemon thread to the end of the queue.

Docket No. AUS920010055US1

5. The method as recited in claim 2, further comprising the steps of:

determining whether the queue is empty; and
if the queue is empty, waiting for the virtual
5 machine to shut down or for a new daemon thread to be created.

6. The method as recited in claim 5, further comprising the step of:

10 if a new daemon thread is created, adding the new daemon thread to the queue.

7. The method as recited in claim 1, wherein the waiter thread is a normal thread.

15 8. The method as recited in claim 1, wherein the daemon threads are created by remote method invocation code.

9. An apparatus for preventing premature shutdown of a
20 virtual machine, comprising:

starting means for starting a waiter thread in the virtual machine;

monitoring means for monitoring daemon threads running in the virtual machine by the waiter thread; and

25 preventing means for preventing shutdown of the virtual machine so long as any of the monitored daemon threads are running in the virtual machine.

10. The apparatus as recited in claim 9, further
30 comprising:

registration means for registering daemon threads in a queue managed by the waiter thread.

0036759-040504
103040-629280

11. The apparatus as recited in claim 10, further comprising:

12. The apparatus as recited in claim 10, further comprising:

13. The apparatus as recited in claim 10, further comprising:

15 determination means for determining whether the
queue is empty; and

20

14. The apparatus as recited in claim 13, further comprising:

means for adding, if a new daemon thread is created, the new daemon thread to the queue.

25

15. The apparatus as recited in claim 9, wherein the waiter thread is a normal thread.

16. The apparatus as recited in claim 9, wherein the
30 daemon threads are created by remote method invocation
code.

17. A computer program product, in a computer readable medium, for preventing premature shutdown of a virtual machine, comprising:

```

    first instructions for starting a waiter thread in
5  the virtual machine;

```

second instructions for monitoring daemon threads running in the virtual machine by the waiter thread; and

```

    third instructions for preventing shutdown of the
    virtual machine so long as any of the monitored daemon
10  threads are running in the virtual machine.

```

18. The computer program product as recited in claim 17,
further comprising:

```

    fourth instructions for registering daemon threads
15  in a queue managed by the waiter thread.

```

19. The computer program product as recited in claim 18,
further comprising:

fifth instructions for responsive to a first daemon
20 ~~thread becoming inactive, searching for other inactive~~
daemon threads registered in the queue.

20. The computer program product as recited in claim 18,
further comprising:

25 sixth instructions for determining whether the queue
is empty; and

seventh instructions for waiting for the virtual machine to shut down or for a new daemon thread to be created if the queue is empty.